nlb

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/693,942	10/28/2003	Kenji Sugiyama	P69233US0	4316

7590 02/28/2007

JACOBSON HOLMAN
PROFESSIONAL LIMITED LIABILITY COMPANY
400 Seventh Street, N.W.
Washington, DC 20004

EXAMINER	
RAO, ANAND SHASHIKANT	
ART UNIT	PAPER NUMBER
2621	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	02/28/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

PTOL-90A (Rev. 10/06)

Office Action Summary	Application No. 10/693,942	Applicant(s) SUGIYAMA, KENJI	
	Examiner Andy S. Rao	Art Unit 2621	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-8* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-8* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *11/24/03*.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

Specification

1. The specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

Claim Rejections - 35 USC § 101

2. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

3. Claims 7-8 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

A). The Examiner notes that "comprising the steps of ..." does not specify how the steps are (a) associated with the medium, or (b) the nature of instructions. Data structures not claimed as embodied (or encoded with or embedded with) in a computer readable medium for a computer implemented method are descriptive material per se, and are not statutory, _Warmerdam,_ *33 F.3d at 1361, 31, USPQ2d at 1760).* Specifying the association in the manner listed above would sufficiently address the first condition. Similarly, computer programs claimed as computer listings, instructions, or codes are just the descriptions, expressions, of the program are not "physical things". They have neither computer components nor statutory processes, as they are not "acts" being performed. In contrast, a claimed "...computer readable medium encoded with a computer program comprising instructions for a computer implemented method..." is a computer element which defines structural and function interrelationships between the computer

program and the rest of the computer, and is statutory, *Lowry, 32 F.3d at 1583-84, 32 USPQ2d at 1035.* Specifying the instructions as a "computer program" would sufficiently address the second condition, *Interim Guidelines, Annex IV (Section a).*

B). Lastly, the computer program as claimed doesn't isn't properly associated with the operation. It is quite possible that the computer program may be an unrelated sub-routine or a simple commence instruction, which then causes the computer to execute the operation that could be self-resident, and not encoded on the medium. The Examiner suggests that the computer program be more directly associated with the operation by further reciting that the "instructions for a computer implemented method, which when executed, cause the computer to execute the steps of..." in order to establish this operative association, *Interim Guidelines, Annex IV (Section b).*

Corrections to the claims, and supporting specification are required.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 1-6 are rejected under 35 U.S.C. 102(b) as being anticipated by Haskell et al., (hereinafter referred to as "Haskell").

Haskell discloses a temporal scalable moving-picture video signal coding method (Haskell: figures 10-11; column 11, lines 34-35), comprising the steps of: converting an input

interlaced moving-picture video signal into a progressive moving-picture video signal at the same frame rate as the interlaced moving-picture video signal (Haskell: figure 39: column 18, lines 5-9); encoding the progressive moving-picture video signal to produce a first bitstream (Haskell: column 10, lines 15-25); encoding fields of the interlaced moving-picture video signal, the fields being different in time from frames of the progressive moving-picture video signal (Haskell: column 9, lines 23-27), with inter- picture prediction using a locally decoded picture signal as a reference video signal (Haskell: column 9, lines 35-40), the locally decoded picture signal being produced by locally decoding the progressive moving- picture video signal, thus producing a second bitstream (Haskell: column 10, lines 35-40); and multiplexing the first and second bitstreams into an output temporal scalable moving-picture video bitstream (Haskell: column 10, lines 45-50), as in claim 1.

Haskell discloses a temporal scalable moving-picture video signal decoding method (Haskell: column 8, lines 65-67) comprising the steps of: demultiplexing a bitstream produced by temporal scalable moving-picture coding into a first bitstream and a second bitstream (Haskell: column 8, lines 55-57), the first bitstream having been produced by encoding a progressive moving-picture video signal at the same frame rate as an interlaced-moving picture video signal to be reproduced (Haskell: column 5, lines 5-10), the second bitstream having been produced, by encoding fields of the interlaced moving-picture video signal, the fields being different in time from frames of the progressive moving-picture video signal (Haskell: column 20, lines 60-67); decoding the first bitstream to reproduce a progressive moving-picture video signal (Haskell: column 19, lines 53-67); converting the reproduced progressive moving-picture video signal into a first field video signal having either even- or odd-number fields of the interlaced moving-

picture video signal (Haskell: column 20, lines 30-45); decoding the second bitstream with inter-picture prediction using the reproduced progressive moving-picture video signal as a reference video signal (Haskell column 19, lines 15-20), thus producing a second field video signal having fields of the interlaced moving-picture video signal, the fields of the second field video signal being different in parity from the fields of the first field video signal (Haskell: column 8, lines 30-43); and switching the first field video signal and the second field video signal to output the interlaced moving-picture video signal (Haskell: column 20, lines 60-65), as in claim 2.

Haskell discloses temporal scalable moving-picture video signal coding apparatus (Haskell: figure 18), comprising : a converter to convert an input interlaced moving- picture video signal into a progressive moving-picture video signal at the same frame rate as the interlaced moving- picture video signal (Haskell: figure 39; column 18, lines 5-9); a first bitstream generator to encode the progressive moving-picture video signal, thus generating a first bitstream (Haskell: column 10, lines 15-25); a second bitstream generator to encode fields of the interlaced moving-picture video signal, the fields being different in time from frames of the progressive moving- picture video signal (Haskell: column 9, lines 23-27), with inter-picture prediction using a locally decoded picture signal as a reference video signal (Haskell: column 9, lines 35-40), the locally decoded picture signal being produced by locally decoding the progressive moving picture video signal, thus producing a second bitstream (Haskell: column 10, lines 35-40); and a multiplexer to multiplex the first and second bitstreams into an output temporal scalable moving-picture video bitstream (Haskell: column 45-50), as in claim 3.

Regarding claim 4, Haskell further discloses a scanning-line down-sampler to which the progressive moving- picture video signal obtained by the converter is supplied (Haskell: column

7, lines 23-27), the down-sampler down-sampling the progressive moving-picture video signal in a spatial vertical direction to produce a progressive moving-picture video signal having a smaller number of scanning lines than the progressive moving-picture video signal obtained by the converter (Haskell: column 7, lines 30-37), wherein the progressive moving-picture video signal having the smaller number of scanning lines is supplied to the first bitstream generator, thus a third bitstream having the smaller number of scanning lines being generated (Haskell: column 7, lines 35-45), and the second bitstream generator has a scanning-line up- sampler to up-sample a locally decoded video signal in the spatial vertical direction (Haskell: column 10, lines 10, lines 20-30), the locally decoded video signal being obtained by locally decoding the third bitstream to produce a video signal having the same number of scanning lines as the progressive moving-picture video signal supplied to the down-sampler, the produced video signal being uses as the reference video signal (Haskell: column 14, lines 1-10), as in the claim.

Haskell discloses a temporal scalable moving-picture video signal decoding apparatus (Haskell: figure 26), comprising: a demultiplexer to demultiplex a bitstream produced by temporal scalable moving-picture coding into a first bitstream and a second bitstream (Haskell: column 8, lines 55-57), the first bitstream having been produced by encoding a progressive moving-picture video sisal at the Same frame rate as an-interlaced moving-picture video signal to be reproduced (Haskell: column 5, lines 5-10), the second bitstream having been produced by encoding fields of the interlaced moving-picture video signal, the fields being different in time from frames of the progressive moving-picture video signal (Haskell: column 20, lines 60-67); a first decoder to decode the first bitstream to reproduce a progressive moving-picture video signal (Haskell: column 19, lines 53-67); a converter to convert the reproduced progressive moving-

picture video signal into a first field video signal having either even- or odd-number fields of the interlaced moving-picture video signal (Haskell: column 20, lines 30-45); a second decoder to decode the second bitstream with inter-picture prediction using the reproduced progressive moving-picture video signal as a reference video signal (Haskell: column 19, lines 15-20), thus producing a second field video signal having fields of the interlaced moving-picture video signal, the fields of the second field video signal being different in parity from the fields of the first field video signal (Haskell: column 8, lines 30-43); and a switch to switch the first field video signal and the second field video signal to output the interlaced moving-picture video signal (Haskell: column 20, lines 60-65), as in claim 5.

Regarding claim 6, Haskell discloses wherein the demultiplexer demultiplexes the bitstream produced by temporal scalable moving-picture coding into the second bitstream and a third bitstream produced by encoding a progressive moving- picture video signal is down-sampled in a spatial vertical direction at the same frame rate as the interlaced moving- picture video signal to be reproduced (Haskell: column 7, lines 23-27), the first decoder decoding the third bitstream into the down-sampled progressive moving-picture video signal and up-sampling the down-sampled and decoded progressive moving-picture video signal in the spatial vertical direction, and the converter converting the up-sampled progressive moving-picture video signal into the first-field-video signal (Haskell: column 14, lines 1-10), as in the claim.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 7-8 are rejected under 35 U.S.C. 103(a) as being unpatentable over Haskell et al., (hereinafter referred to as "Haskell") in view of Van Der Schaar.

Haskell discloses a temporal scalable moving-picture video signal coding method (Haskell: figures 10-11; column 11, lines 34-35), comprising the steps of: converting an input interlaced moving-picture video signal into a progressive moving-picture video signal at the same frame rate as the interlaced moving-picture video signal (Haskell: figure 39: column 18, lines 5-9); encoding the progressive moving-picture video signal to produce a first bitstream (Haskell: column 10, lines 15-25); encoding fields of the interlaced moving-picture video signal, the fields being different in time from frames of the progressive moving-picture video signal (Haskell: column 9, lines 23-27), with inter- picture prediction using a locally decoded picture signal as a reference video signal (Haskell: column 9, lines 35-40), the locally decoded picture signal being produced by locally decoding the progressive moving- picture video signal, thus producing a second bitstream (Haskell: column 10, lines 35-40); and multiplexing the first and second bitstreams into an output temporal scalable moving-picture video bitstream (Haskell: column 10, lines 45-50), as in claim 7. However, Haskell fails to disclose that the method is a computer implemented method as in the claim. Van Der Schaar discloses a method for scaleable encoding (Van Der Schaar: figures 6-7) which is executed with computer implementation (Van Der Schaar: column 9, lines 60-67) in order to apply the method to distributed networks (Van Der Schaar: column 9, lines 45-55). Accordingly, given this teaching it would have been obvious for one of ordinary skill in the art incorporate the Van Der Schaar teaching of computer

implementation to the Haskell encoding method in order to have the Haskell method applicable to distributed networks. The Haskell encoding method, now modified to be executable by computer implementation as shown by Van Der Schaar, has all of the features of claim 7.

Haskell discloses a temporal scalable moving-picture video signal decoding method (Haskell: column 8, lines 65-67) comprising the steps of: demultipiexing a bitstream produced by temporal scalable moving-picture coding into a first bitstream and a second bitstream (Haskell: column 8, lines 55-57), the first bitstream having been produced by encoding a progressive moving-picture video signal at the same frame rate as an interlaced-moving picture video signal to be reproduced (Haskell: column 5, lines 5-10), the second bitstream having been produced, by encoding fields of the interlaced moving-picture video signal, the fields being different in time from frames of the progressive moving-picture video signal (Haskell: column 20, lines 60-67); decoding the first bitstream to reproduce a progressive moving-picture video signal (Haskell: column 19, lines 53-67); converting the reproduced progressive moving-picture video signal into a first field video signal having either even- or odd-number fields of the interlaced moving-picture video signal (Haskell: column 20, lines 30-45); decoding the second bitstream with inter-picture prediction using the reproduced progressive moving-picture video signal as a reference video signal (Haskell column 19, lines 15-20), thus producing a second field video signal having fields of the interlaced moving-picture video signal, the fields of the second field video signal being different in parity from the fields of the first field video signal (Haskell: column 8, lines 30-43); and switching the first field video signal and the second field video signal to output the interlaced moving-picture video signal (Haskell: column 20, lines 60-65), as in claim 8.
However, Haskell fails to disclose that the method is a computer implemented method as in the

claim. Van Der Schaar discloses a method for scaleable encoding (Van Der Schaar: figures 6-7) which is executed with computer implementation (Van Der Schaar: column 9, lines 60-67) in order to apply the method to distributed networks (Van Der Schaar: column 9, lines 45-55). Accordingly, given this teaching it would have been obvious for one of ordinary skill in the art incorporate the Van Der Schaar teaching of computer implementation to the Haskell decoding method in order to have the Haskell method applicable to distributed networks. The Haskell decoding method, now modified to be executable by computer implementation as shown by Van Der Schaar, has all of the features of claim 8.

Conclusion

8. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Andy S. Rao whose telephone number is (571)-272-7337. The examiner can normally be reached on Monday-Friday 8 hours.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Mehrdad Dastouri can be reached on (571)-272-7418. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Andy S. Rao
Primary Examiner
Art Unit 2621

ANDY RAO
PRIMARY EXAMINER

asr
February 26, 2007